CLCS System Test Plan

April 23, 1997

CLCS

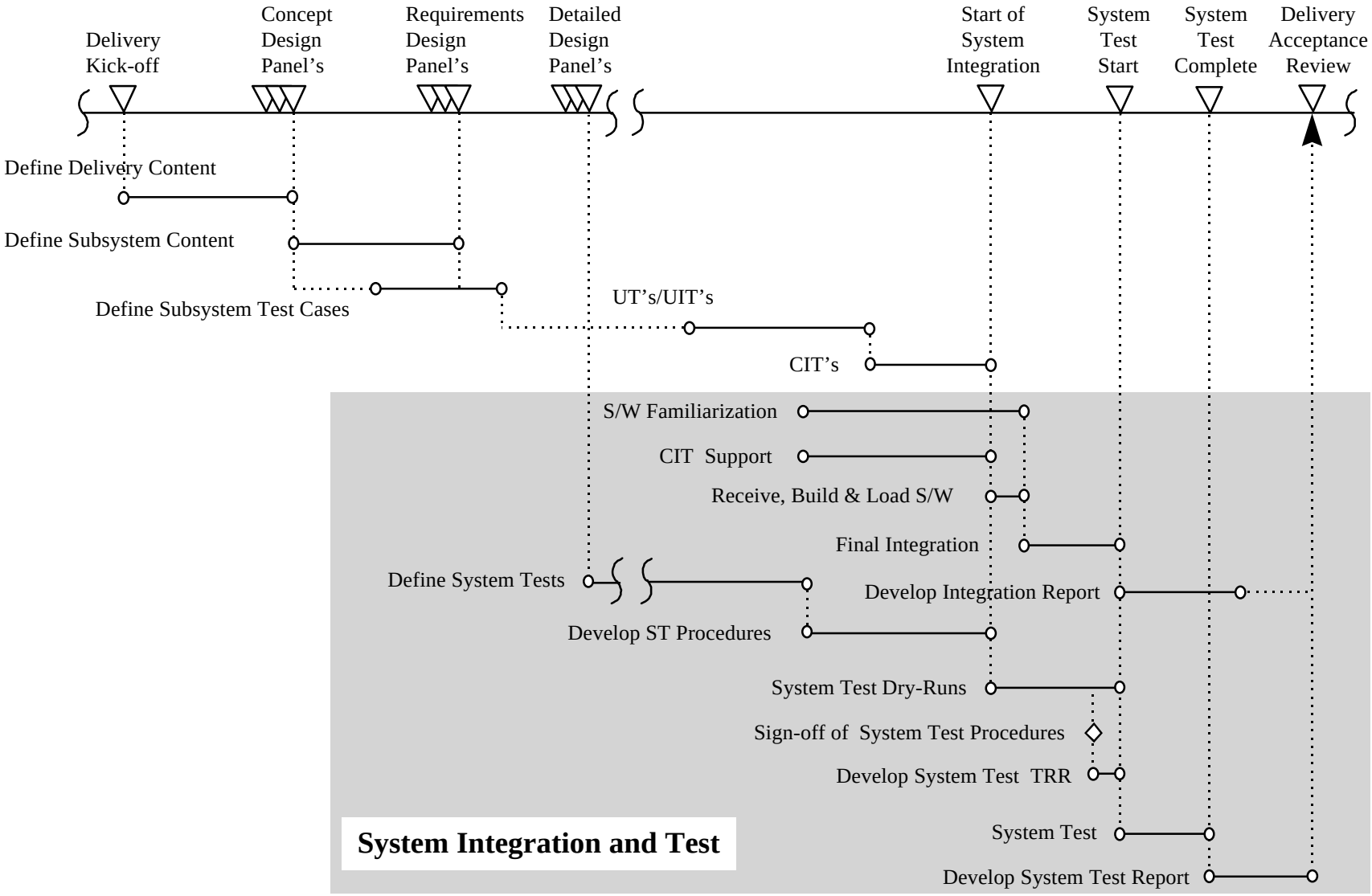
Agenda

- Baselined CLCS Test Process Summary
  - CLCS Test Approach
  - Requirements Verified by Tests
  - Test Definitions
  - Certification Philosophy Overview
- Next Steps

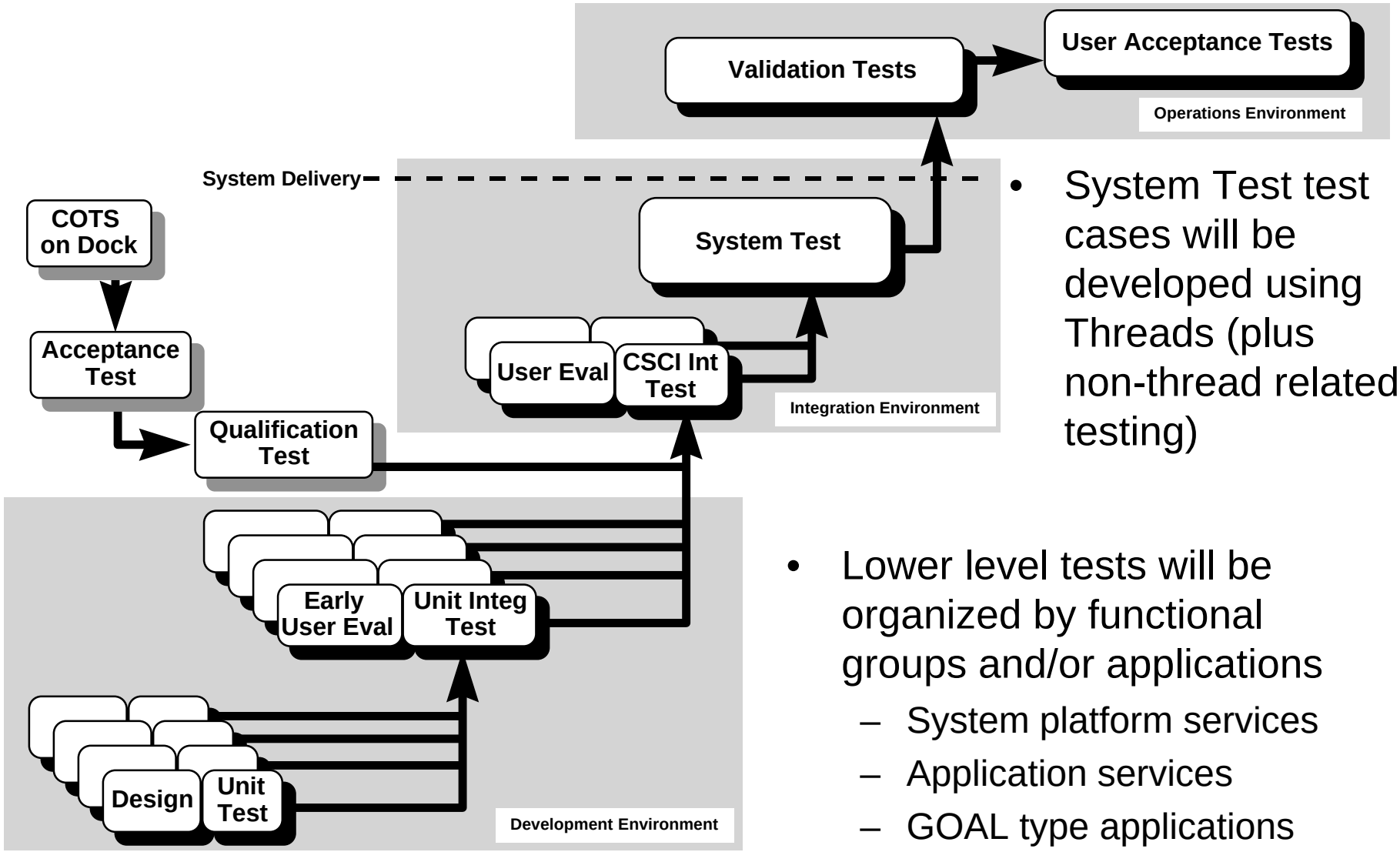
CLCS Test Approach

- Each test is developed and reviewed in relation to previous testing
  - Subsystem and system level tests do not simply repeat procedures performed in previous tests, emphasis is placed on what needs to be tested in the context of each level of test
  - Tests from previous deliveries are modified as necessary and reused where possible
- Early user evaluation will be an integral part of the test development process
  - Feed back from user experience with early versions of software will allow for the development of more focused tests (in the higher level tests)
    - Areas of concern to the user community will be emphasized
    - Coordination of the content of different levels of testing will be easier
- End-to-end testing process will be consistent across all software types and development organizations
  - Re-engineered “GOAL type” applications, System/Platform Services, application services
  - NASA, USA, LMSMS

Integration & Test Delivery Schedule Template



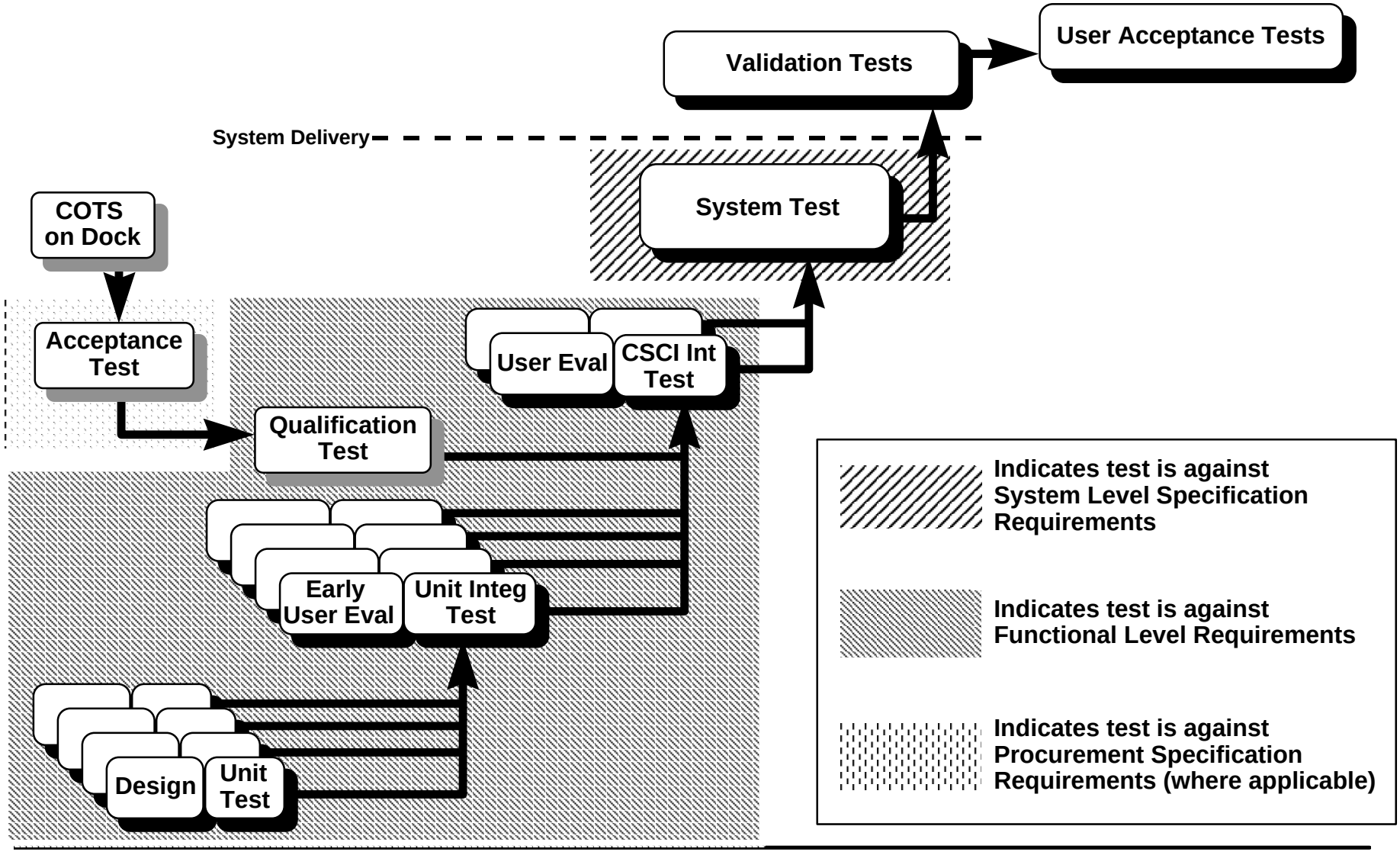
CLCS Test Approach (One Delivery Cycle Shown)



- System Test test cases will be developed using Threads (plus non-thread related testing)

- Lower level tests will be organized by functional groups and/or applications
  - System platform services
  - Application services
  - GOAL type applications

Requirements Verified by Tests



Test Definitions

- Unit Test (UT) is performed by S/W developer to verify basic functionality of a demonstrable segment of a CSCI
  - The level of formality (amount and type of documentation) is minimal
  - The tests are signed off by the development lead, UT's are witnessed and signed-off by Software QA (S&MA) only when the test is formally demonstrating compliance with Functional Requirements
- Unit Integration Test (UIT) is performed by S/W developer to verify basic functionality and successful integration of a set of programs (CSCI) with its new and/or modified units
  - UIT's are witnessed and signed-off by S&MA when the test is formally demonstrating compliance with Functional Requirements

Test Definitions (cont'd)

- CSCI Integration Test (CIT) is performed by S/W developer and/or thread lead to verify basic functionality and successful integration of a functionally complete set of programs (CSCI's, e.g., GLS, LOX, System Services) with its new and/or modified units in an operational environment
  - Low level regression testing performed
  - CIT's are witnessed and signed-off by S&MA
- System Test (ST) is performed to verify successful integration of a system delivery and to demonstrate that the system is "operable" in all required modes (development, maintenance, system operations, test/launch support, etc.)
  - This is the final level of testing prior to putting the system in an operations
  - System level regression testing performed
  - ST's are witnessed and signed-off by S&MA

Test Definitions (cont'd)

- Validation tests are performed by an independent group at KSC (independent of the developing organization)
  - Verify that the product (a delivery) handed over to the Ops organization from the development organization meets the users requirements
  - Executed for each system delivery
- User Acceptance tests verify that the system is processing data correctly (data and displays are valid) and that the system can be certified for use in approved types of operations (e.g., SLWT, HMF, Launch)
  - Scope of tests defined by user community on a per test basis
  - Executed in the ops environment as required by users, usually in preparation for operational certification
  - This is the final level of test prior to full operations support
- The NASA Software Independent Verification & Validation Facility at Fairmont West Virginia will be supporting CLCS; plans, approach and schedule are to be determined.

Roles and Responsibilities

Test	Development Organization	System Integration Group	System Test Group	End User Community
Unit Test	R			
Unit Integration test	R			
CSCI Integration Test	R	S		
CIT Functional Testing	R	S		
CIT Regression Testing	S	S	R	
System Test		S	R	
Validation Test			S	R
User Acceptance Test			S	R
COTS				
Acceptance Test	R			
Qualification Test	R	S		

R = Responsible, S = Support

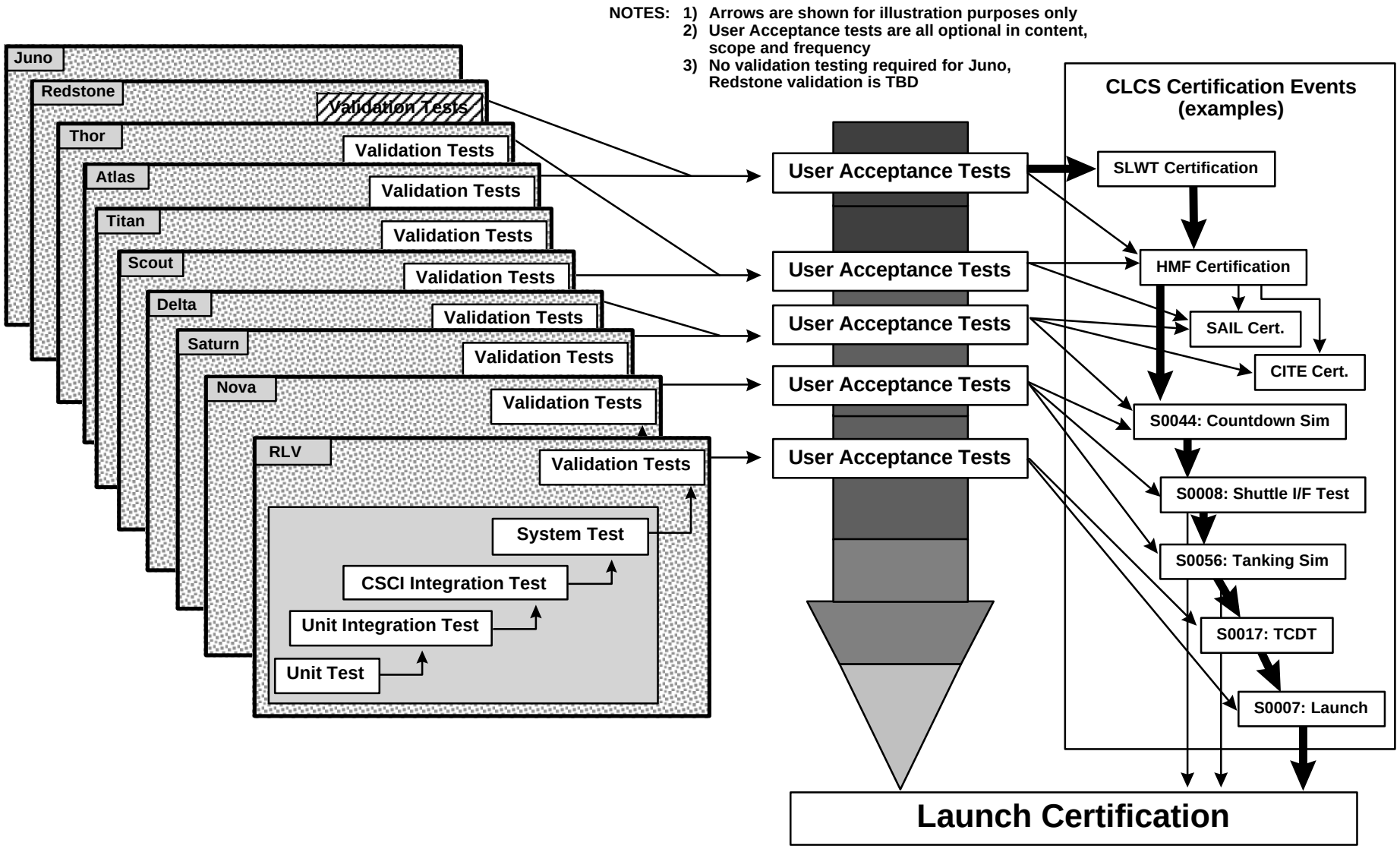
- The chart to the left shows the roles of organizations regarding the different tests

- The chart to the right shows the roles of organizations regarding the System Tests

Organization	System Test Plan	System Test Procs.	ST Dry Run	TRR	ST Exec.	TPR	PTR	Test Report
CLCS Developers	R	R	S					
CLCS System Integration	R	R	S,E	S	S	S	S	R
CLCS System Test	G	G	E	G,E	E	G,E	G,E	G
CLCS System Admin		R	S	S	S	S	S	
NASA S&MA	R,A	R,A		R,A	W	R,A	R,A	R,A
USA S&MA	R	R		R		R	R	R
LMSMS QA	R	R						R
End User Community	R	R		S	S	S	R	R
Operations Support			S	S	S	S	S	

W = witness (with signature approval), E = execute, R = review/attend (comments),
A = approve (signature), S = support, G = generate

Certification Philosophy Overview



Next Steps

- Proceed with the baselined integration and test process for the Redstone delivery
- Work schedule coordination concerns:
  - Math Model/SGOS development and validation
  - Identify what test resources are needed (automated tools, testers, users, sims/models)
- Refine plans for regression testing, including development of regression test tools/procedures
- Coordinate with software developers on details of Application and System Services development and verification approach
- Refine process for tracing requirements to threads, CSCI's and tests